Amendment to the Specification

[0010] FIG. 1 ~~is. Figure one~~ is a data and process flow diagram showing a distributed computing system 100, in which embodiments of the present invention are envisioned to be useful.

[0029] Another situation in which an application may be at risk occurs when a computationally expensive program is to be broken into pieces to be processed by numerous computers. An example of such computing is a screen saver computation program, where a large application or an application analyzing a large body of data is spread over thousands of personal computers to be executed by screen saver programs that operate when the user is not otherwise employing the processor. Yet another situation ~~such situation~~ occurs when an application is storing data on a potentially insecure machine, whether or not the application itself is executing on a trusted machine or machines. As should be clear from the foregoing description, an application using sensitive data nearly always places that data at some level of risk.

[0058] In performing a read operation, user program 602 calls read function 614. Read function 614 first, in step 640, reads a segment of encrypted data 634 from encrypted, temporary file 632. The data subset 634 is then decrypted by a function local to application ~~601~~ 602 (meaning on application side 600) at step 642. The decrypted data is then authenticated using the digital signature of encrypted, temporary file 632, and if successful, used by user program 602.

[0145] Accordingly, a second example embodying the invention is presented in the CD appendix ~~at files~~ _____. This embodiment is a general-purpose embedded database library operating securely using the DAP embodiments (DAPDB). The DAPDB database is linked with the user application and resides in the same address space as the user application. The DAPDB is implemented in the form of a function library (or "library"). The database is constructed such that each table is associated to a single DAP file (encrypted, checked, signed); such that the database is not decrypted in memory for obvious security reasons; such that each record in the table is composed by a record key (not to be confused with an encryption key, a decryption key or a user key) and a record body string; for each record the record key and the record body string can have arbitrary size and can contain non-homogeneous structures (for example record bodies can contain strings of arbitrary length or any user defined data structures, in the same fashion the

record key can be any user defined data structure); such that a fast record key search that does not require record key decryption or creating temporary index can be used; such that the records in the database can be appended, deleted and replaced; such that the records in the database can be accessed in the order they were appended or modified (The last appended/modified record always appears as the last record in the database) and such that the database is transaction safe (for operations on a single table). In the present embodiment each data table is associated to a hash table of checksums of record keys. Each hash table is stored, signed and encrypted together with the corresponding database table in a DAP file. Once a table is opened the hash table is decrypted and stored in memory for fast search.